

Centro Universitário UniSenac – Campus Pelotas

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Desenvolvimento de Serviços e APIs – Trabalho #2

Nome: Bernardo Carvalho

Descrição do Projeto: O sistema da Locadora Avenida permite a inclusão, listagem e exclusão de novos clientes, filmes e locações, além de enviar um e-mail para um cliente com seu histórico de locações.

Enviei e-mail com a descrição e modelo E-R no dia 30/05: Sim

## 1. Rotas

```
// @ts-ignore
import { swaggerUi, specs } from "./swagger";

import express from "express";
import routesClientes from "./routes/clientes";
import routesFilmes from "./routes/filmes";
import routesLocacoes from "./routes/locacoes";

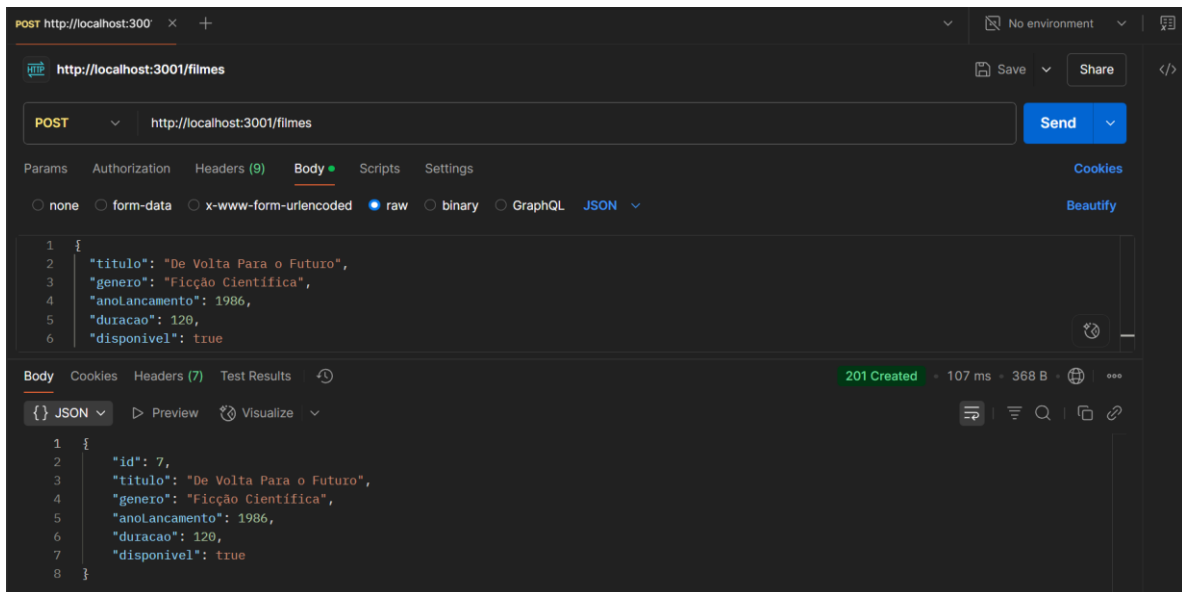
const app = express();
const port = 3001;

app.use(express.json());
app.use("/api-docs", swaggerUi.serve, swaggerUi.setup(specs));
app.use("/clientes", routesClientes);
app.use("/filmes", routesFilmes);
app.use("/locacoes", routesLocacoes);

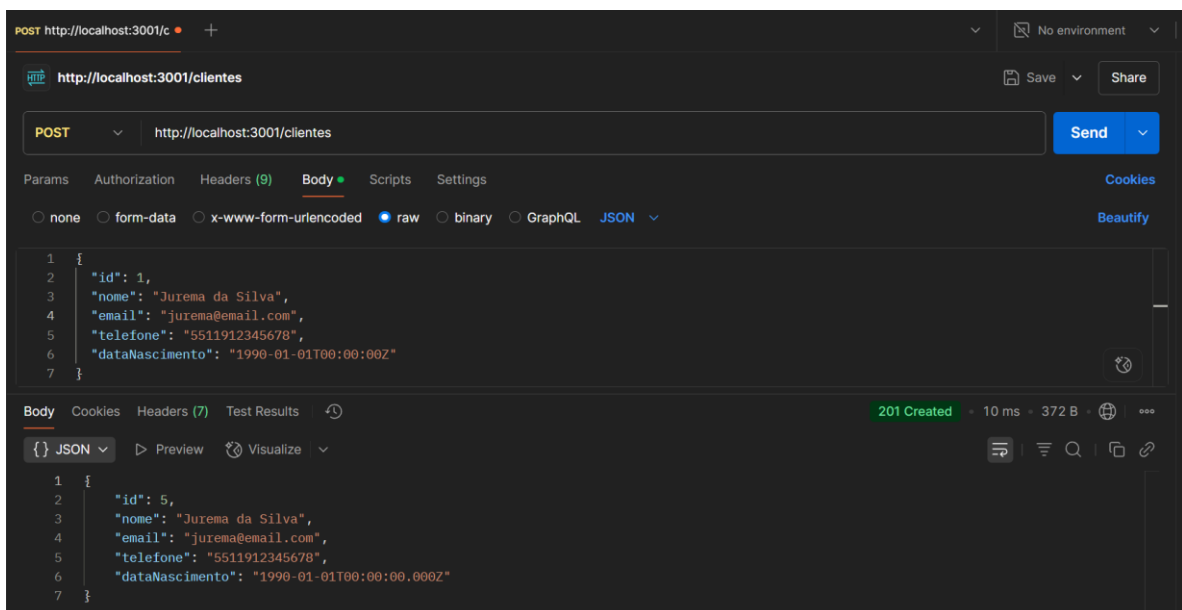
app.get("/", (req: any, res: any) => {
  res.send("API: Locadora Avenida");
});

app.listen(port, () => {
  console.log(`Rodando na porta: ${port}`);
});
```

## 2. Inclusão de Filmes



### 3. Inclusão de Clientes



\* Desconsiderar id = 1 na requisição, visto que a tabela utiliza auto\_increment para o campo id.

### 4. Código da Inclusão da Transação

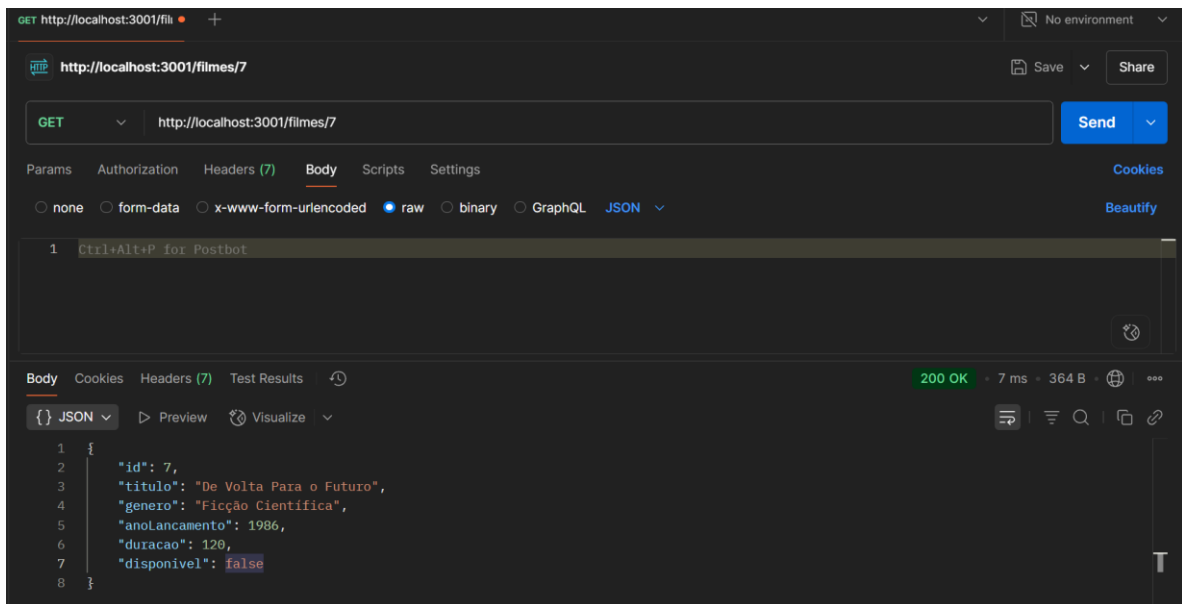
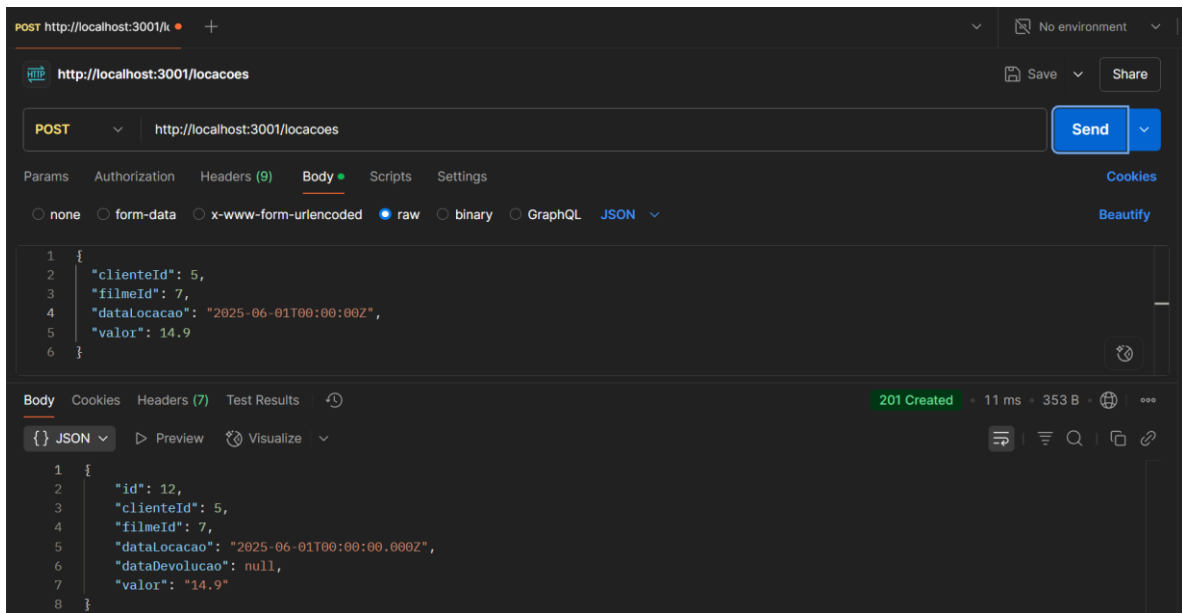
```
const [_, locacao] = await prisma.$transaction([
  prisma.filme.update({
    where: { id: filmeId },
    data: { disponivel: false },
  }),
  prisma.locacao.create({
    data: {
      clienteId,
      filmeId,
      dataLocacao,
      dataDevolucao,
      valor,
    },
  }),
]);
```

## 5. Demonstração da Transação

The screenshot shows a web browser interface with a GET request to `http://localhost:3001/filmes/7`. The response is a JSON object with the following details:

```
{
  "id": 7,
  "titulo": "De Volta Para o Futuro",
  "genero": "Ficção Científica",
  "anoLancamento": 1986,
  "duracao": 120,
  "disponivel": true
}
```

The status bar at the bottom indicates a **200 OK** response with a time of 12 ms and a size of 363 B.



6. Código da Exclusão da Transação (exclusão de locação e cliente, respectivamente)

```
const locacao = await prisma.locacao.findUnique({
  where: {
    id: Number(id),
  },
});

if (!locacao) {
  return res.status(404).json({ erro: "Locação não encontrada" });
}

await prisma.$transaction([
  prisma.locacao.delete({
    where: {
      id: Number(id),
    },
  }),
  prisma.filme.update({
    where: {
      id: locacao.filmeId,
    },
    data: {
      disponivel: true,
    },
  }),
]);
```

```

router.delete("/:id", async (req, res) => {
  try {
    const { id } = req.params;

    await prisma.$transaction(async (tx) => {
      const locacoes = await tx.locacao.findMany({
        where: { clienteId: Number(id) },
        select: { filmeId: true },
      });

      const filmeIds = locacoes.map((l) => l.filmeId);

      await tx.filme.updateMany({
        where: { id: { in: filmeIds } },
        data: { disponivel: true },
      });

      await tx.locacao.deleteMany({
        where: { clienteId: Number(id) },
      });

      await tx.cliente.delete({
        where: { id: Number(id) },
      });
    });

    res.status(200).json(id);
  }
});

```

7. Demonstração da Transação (exclusão de locação e cliente, respectivamente)

GET http://localhost:3001/loc

http://localhost:3001/locacoes

GET http://localhost:3001/locacoes

Params Authorization Headers (7) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

1 Ctrl+Alt+P for Postbot

Body Cookies Headers (7) Test Results

200 OK 10 ms 350 B

JSON Preview Visualize

```
1 [
2   {
3     "id": 12,
4     "clienteId": 5,
5     "filmeId": 7,
6     "dataLocacao": "2025-06-01T00:00:00.000Z",
7     "dataDevolucao": null,
8     "valor": "14.9"
9   }
10 ]
```

DEL http://localhost:3001/loc

http://localhost:3001/locacoes/12

DELETE http://localhost:3001/locacoes/12

Params Authorization Headers (7) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

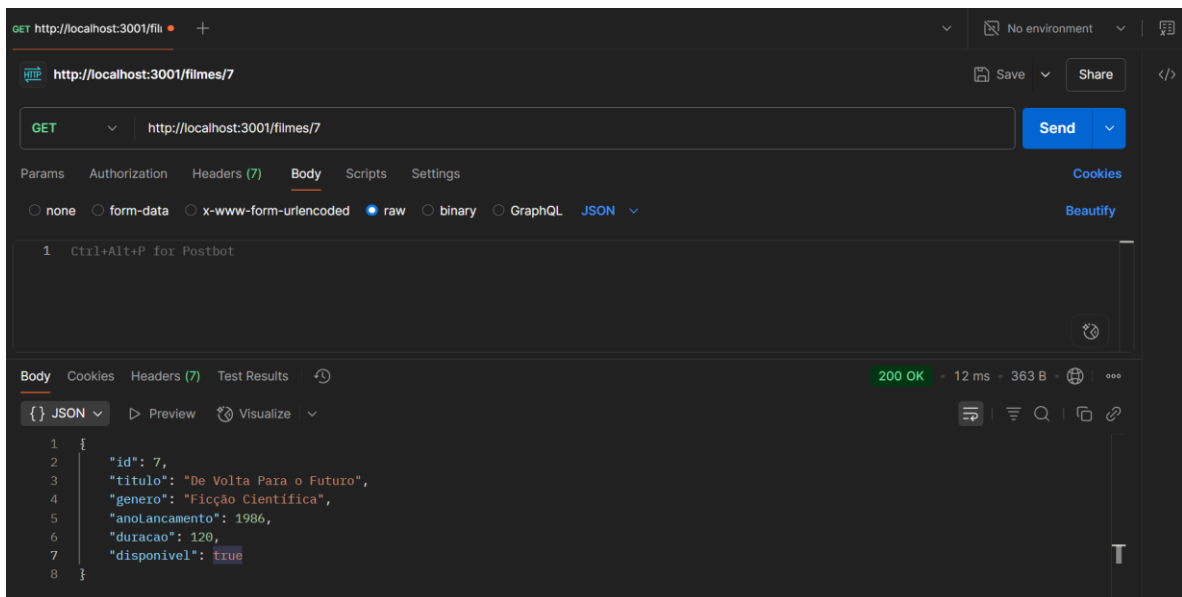
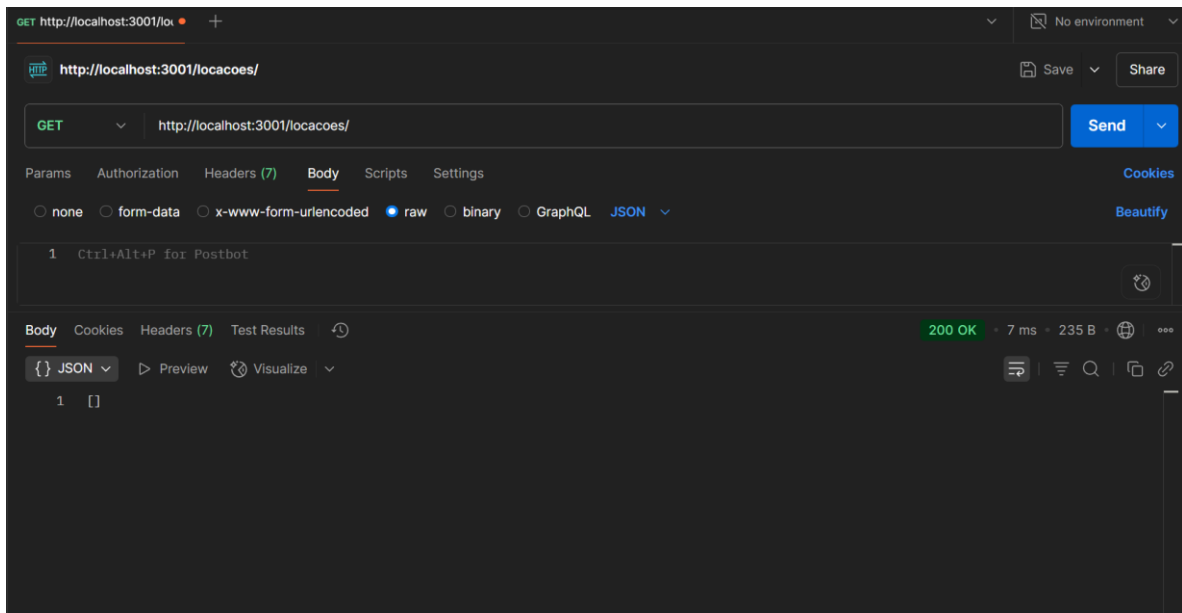
1 Ctrl+Alt+P for Postbot

Body Cookies Headers (7) Test Results

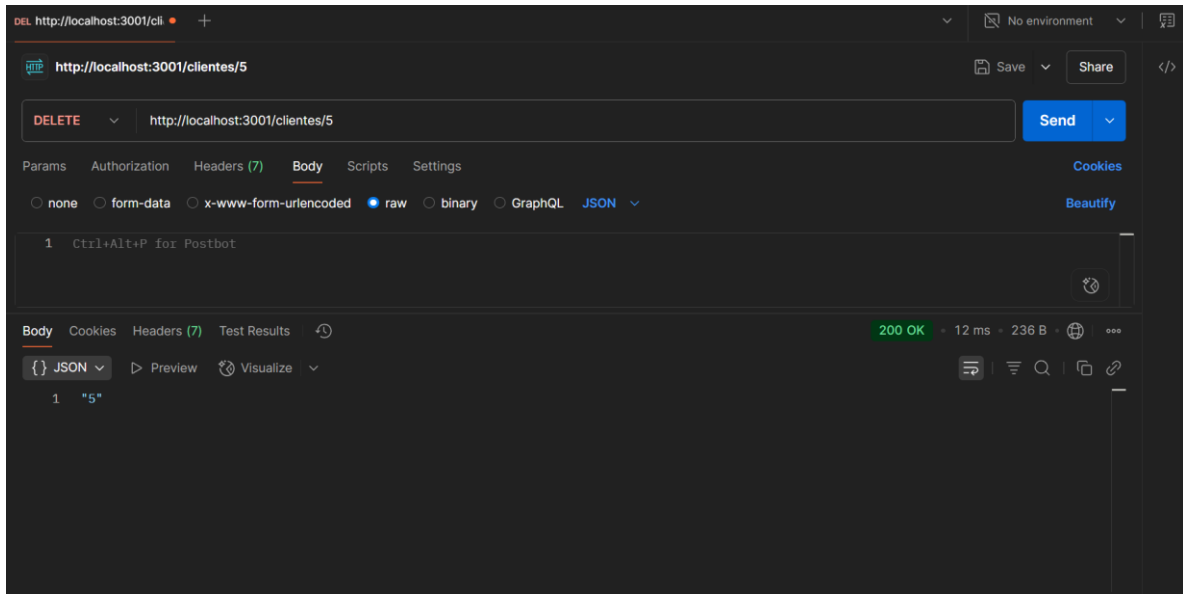
200 OK 13 ms 302 B

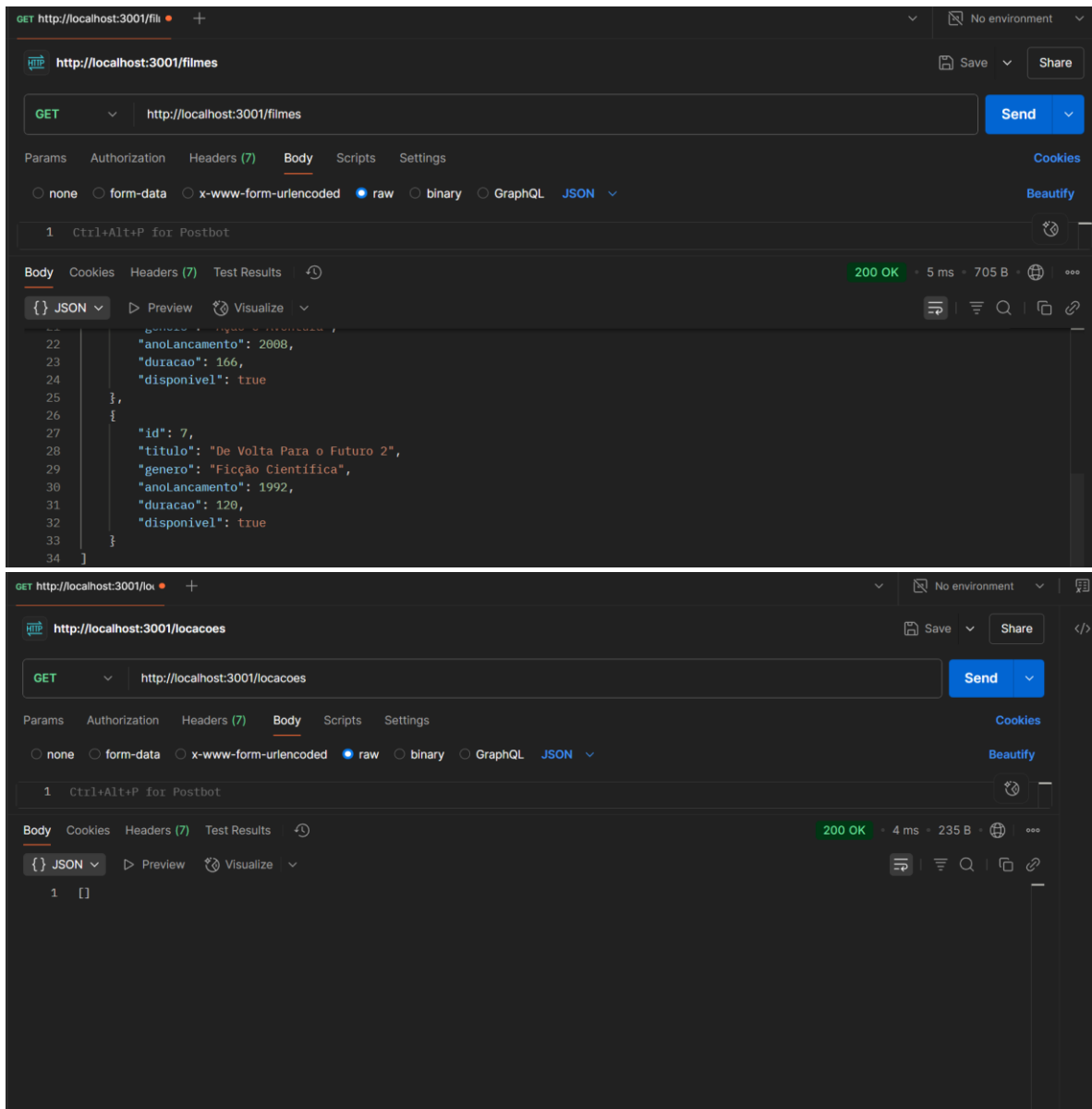
JSON Preview Visualize

```
1 {
2   "mensagem": "Locação removida e filme disponibilizado",
3   "id": "12"
4 }
```









8. Demonstração de alteração dos dados (clientes e filmes respectivamente)

```

router.put("/:id", async (req, res) => {
  try {
    const result = clientesSchema.safeParse(req.body);

    if (!result.success) {
      res.status(400).json({ erro: "Informe todos os campos corretamente" });
      return;
    }

    const { nome, email, telefone, dataNascimento } = result.data;
    const { id } = req.params;

    const cliente = await prisma.cliente.update({
      data: {
        nome,
        email,
        telefone,
        dataNascimento,
      },
      where: {
        id: Number(id),
      },
    });

    res.status(200).json(cliente);
  } catch (error) {
    res.status(500).json({ erro: error });
  }
});

```

PUT http://localhost:3001/ × + No environment

http://localhost:3001/clientes/5 Save Share

PUT http://localhost:3001/clientes/5 Send

Params Authorization Headers (9) Body Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```

1 {
2   "nome": "Jurema Da Silva Sauro",
3   "email": "jurema_sauero@email.com",
4   "telefone": "5511912345678",
5   "dataNascimento": "1990-01-01T00:00:00Z"
6 }

```

Body Cookies Headers (7) Test Results 200 OK 26 ms 379 B

{ JSON Preview Visualize

```

1 {
2   "id": 5,
3   "nome": "Jurema Da Silva Sauro",
4   "email": "jurema_sauero@email.com",
5   "telefone": "5511912345678",
6   "dataNascimento": "1990-01-01T00:00:00.000Z"
7 }

```

```

router.put("/:id", async (req, res) => {
  try {
    const result = filmesSchema.safeParse(req.body);

    if (!result.success) {
      res.status(400).json({ erro: "Informe todos os campos corretamente" });
      return;
    }

    const { titulo, genero, anoLancamento, duracao, disponivel } = result.data;
    const { id } = req.params;

    const filme = await prisma.filme.update({
      data: {
        titulo,
        genero,
        anoLancamento,
        duracao,
        disponivel,
      },
      where: {
        id: Number(id),
      },
    });

    res.status(200).json(filme);
  } catch (error) {

```

PUT http://localhost:3001/filmes/7

PUT http://localhost:3001/filmes/7

Params Authorization Headers (9) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```

1 {
2   "titulo": "De Volta Para o Futuro 2",
3   "genero": "Ficção Científica",
4   "anoLancamento": 1992,
5   "duracao": 120,
6   "disponivel": true
7 }

```

Body Cookies Headers (7) Test Results

200 OK • 8 ms • 365 B

JSON Preview Visualize

```

1 {
2   "id": 7,
3   "titulo": "De Volta Para o Futuro 2",
4   "genero": "Ficção Científica",
5   "anoLancamento": 1992,
6   "duracao": 120,
7   "disponivel": true
8 }

```

9. Demonstração do Envio do e-mail

POST http://localhost:3001/c

http://localhost:3001/clientes/enviar-relatorio/5

POST http://localhost:3001/clientes/enviar-relatorio/5

Send

Params Authorization Headers (8) Body Scripts Settings

Body Cookies Headers (7) Test Results

200 OK · 2.41 s · 696 B

{ } JSON Preview Visualize

```
2  "mensagem": "Email enviado com sucesso",
3  "cliente": {
4    "id": 5,
5    "nome": "Jurema Da Silva Sauro",
6    "email": "jurema_sauro@email.com",
7    "telefone": "5511912345678",
8    "dataNascimento": "1990-01-01T00:00:00.000Z",
9    "locacoes": [
10     {
11       "id": 13,
12       "clienteId": 5,
13       "filmeId": 7,
14       "dataLocacao": "2025-06-01T00:00:00.000Z",
15       "dataDevolucao": null,
16       "valor": "14.9",
17       "filme": {
18         "id": 7,
19         "titulo": "De Volta Para o Futuro 2".
```

Seu Relatório de Locações

From: Locadora <locadora-avenida@no-reply.com> To: <jurema\_sauro@email.com> 2025-06-06 23:14, 1.9 KB

Show Headers

HTML HTML Source Text Raw Spam Analysis HTML Check 1 Tech Info

Locadora Avenida - Relatório de Locações

Cliente: Jurema Da Silva Sauro

Email: jurema\_sauro@email.com

Data da Locação	Título do Filme	Gênero	Ano	Duração (min)	Data de Devolução	Valor (R\$)
31/05/2025, 21:00	De Volta Para o Futuro 2	Ficção Científica	1992	120	Não devolvido	14,90
Total:						R\$ 14,90

10. Print do modelo ER (Tabelas do Sistema)

